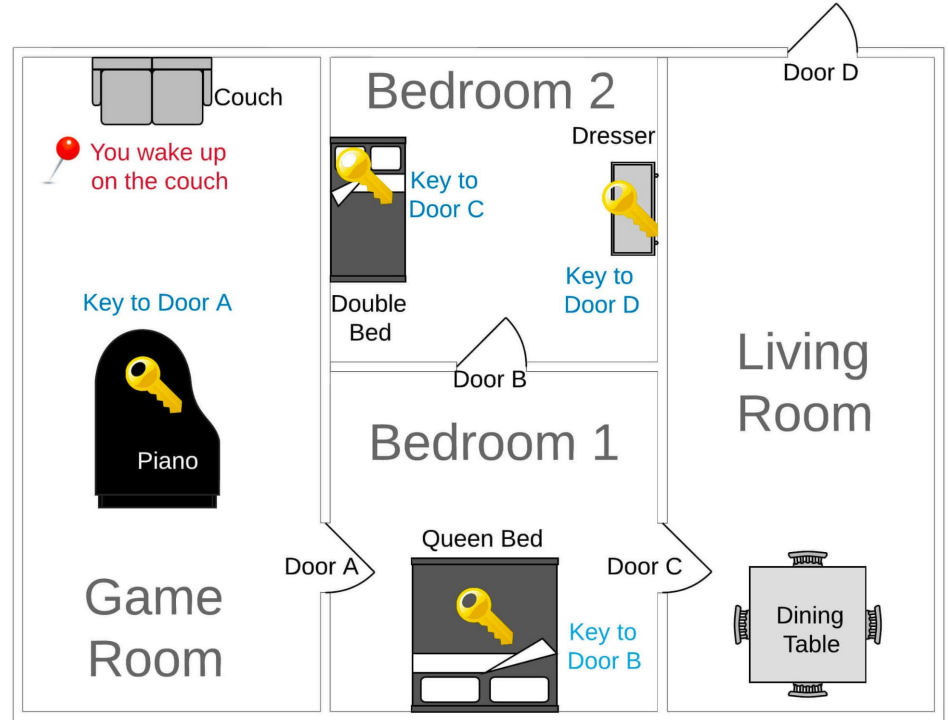


The Dragon Key

Green Team

Game Narrative & Deliverable


1. Develop the game using Python based on the game design.
2. Expand the code following the example to create Bedroom 1.
3. Improve the MVP with New Features




Sales Speech with Demo

 *All-New Escape Room Experience!* 

Are you ready for the ultimate challenge?

 ***New Rooms and Doors:** Uncover secrets and hidden treasures in never-before-seen settings. With new rooms, new doors, and new enigmas, you'll be transported to a world of adventure and excitement!*

 ***New Puzzles:** Prepare to be amazed by our puzzles that challenge your logic, teamwork, and problem-solving skills. Can you crack the code, find the clues and escape?*

Don't miss out on this thrilling opportunity to put your skills to the test. Book your adventure and prepare for an unforgettable experience!

Problem Solving Approach

Game Initialization:

1. Define all rooms, furniture, doors, and keys.
2. Define the relations between objects.
3. Initialize game state.

Pseudocode:

1. Print game's introductory message.
2. Determine the current room of the player.
3. If the current room is the target room (Outside), end the game with a success message.
4. Otherwise, ask the player for an action (explore or examine).
 - a. If action is "explore":
 - i. List all items in the current room.
 - ii. Return to step 2.
 - b. If action is "examine":
 - i. Ask which item the player wants to examine.
 - ii. If the item is in the current room:
 1. If the item is a key, add it to player's keys_collected.
 2. If the item is a door and player has the key, open the door and move to the connected room.
 3. If the item is a door and player doesn't have the key, notify player that the door is locked.
 4. Return to step 2.
 - iii. Else, notify the player that the item isn't in the current room and return to step 2.

Technicalities

```
object_relations = {
    "game room": [couch, piano, door_a],
    "bedroom 1": [queen_bed, door_b, door_c],
    "bedroom 2": [double_bed, dresser, door_b],
    "living room": [door_c, dining_table, door_d, dragon_door, black_couch, tv],
    "bathroom": [door_d, mirror],
    "outside": [dragon_door],

    "piano": [key_a],
    "queen bed": [key_b],
    "double bed": [key_c],
    "dresser": [key_d],
    "dining table": [dragon_key],

    "door a": [game_room, bedroom_1],
    "door b": [bedroom_1, bedroom_2],
    "door c": [bedroom_1, living_room],
    "door d": [living_room, bathroom],
    "dragon door": [living_room, outside]
}
```

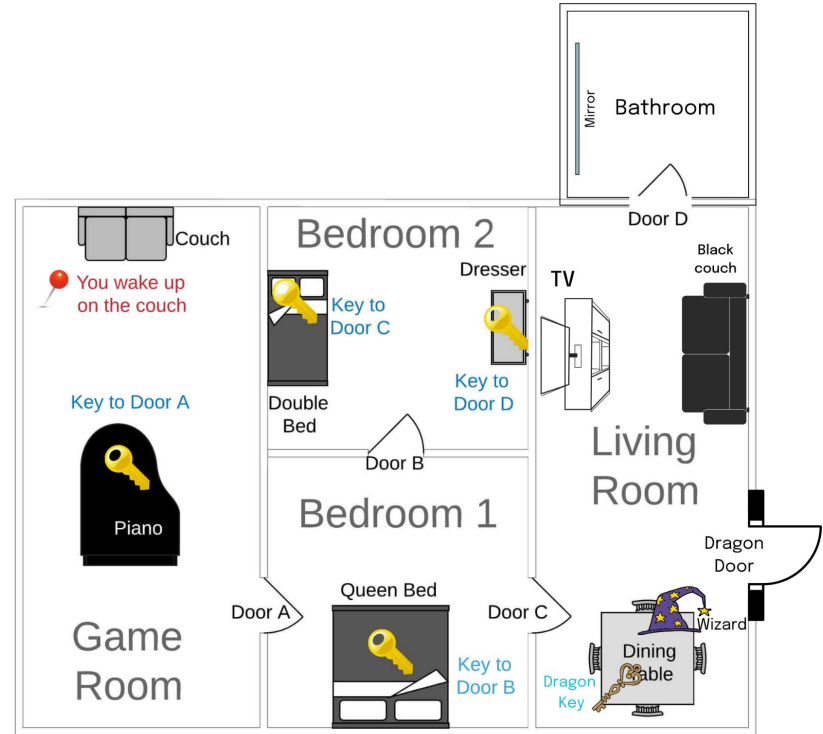
```
elif(item["name"] == "dining table"):
    output_riddle = ""
    You have found a weird wizard hidden under the table...\n
    It seems he has something for you, but you should answer his question first:\n
    I speak without a mouth and hear without ears. I have no body, \n
    but I come alive with the wind.""
    print(output_riddle)

while True:
    riddle_answer = input("What am I? ").strip().lower()
    if riddle_answer == "echo":
        item_found = object_relations[item["name"]].pop()
        game_state["keys_collected"].append(item_found)
        output = "You find " + item_found["name"] + "."
        break
    else:
        print("You should think more carefully...")

elif(item["name"] == "mirror"):
    output_riddle2 = ""
    In the mirror, four jumbled letters appear...\n
    I don't know what they might mean...\n
    H O C E ""
    print(output_riddle2)
```

New Features

- More Rooms:
 - Bathroom
- More Doors:
 - Dragon door
- More Keys:
 - Dragon key
- More Furniture:
 - TV
 - Black Couch
 - Mirror
- Interactions:
 - Quizzes (x2)
- Characters
 - Wizard



Future Improvements

1. Different game modes:
 - Easy
 - Advanced
 - Legend
2. Multiplayer mode
3. Import and display images to add more interactive riddles
4. Doors locked by 2 or more padlocks

Project Management

Team Member	Role	Description
Whitney	Project Manager	Oversaw the project timeline, coordinated meetings, and ensured all deliverables were met on time.
Jonatan	Developer	Focused on writing, testing, and debugging the Python code for the project.
Pablo	Documentation & Repo Manager	Managed the GitHub repository, ensured that the project was properly documented, and submitted the repository link as required.
Osagie	Presentation Lead	Prepared the slides, rehearsed the presentation, and ensuring that it adhered to the given time constraints.